INTERNATIONAL CONFERENCE ON ENGINEERING DESIGN
ICED 05 MELBOURNE, AUGUST 15-18, 2005

DESIGN REPRESENTATIONS FOR MULTIPLE-VIEWPOINT ANALYSIS WITH A FOCUS ON CONCEPTUAL AIRCRAFT DESIGN

D. Davies and C. A. McMahon

Keywords: feature-based design, viewpoint dependency, design automation, conceptual design, aircraft design

1. Introduction

The aim of the work reported here is to investigate ways of representing designed objects that facilitate the evaluation of the objects from multiple perspectives or viewpoints throughout the design process from the concept phase through to detail. The exemplar application that is being used in the work is the design of aircraft in undergraduate student projects.

The paper will review current approaches to multiple-viewpoint modelling in design, especially parametric and feature-based methods, with an emphasis on the limitations of these approaches and the implications of these limitations for design activities. A new approach to modelling, based on applying ideas of semantic mark-up to computer-aided engineering models, will then be presented, and a scenario for application of the approach to aircraft design in undergraduate design projects will be described.

2. Issues in Design Representation

During the design process, various engineering representations of the designed artefact are created and used for different engineering tasks [1]. These representations are used to capture and evaluate different modelled properties of the artefact. These modelled properties may be divided into two main classes: design parameters that describe the structure, shape, dimension, surface finish and other attributes of the product for subsequent manufacture, typically represented in drawings, diagrams and computer-aided design (CAD) models; and performance parameters that describe the characteristics and behaviour of the products subjected to the operating environment that applies loads and other boundary conditions to the product [2]. The design parameter models are typically what the design engineer produces. These models are used, together with information about the way the artefact may be used or made, to estimate the performance or the artefact and to prepare information concerning its manufacture. Additional models – including computer models, mathematical models, graphical models and physical prototypes and test pieces – will assist in these tasks, which may be regarded as evaluating or processing the design from different viewpoints.

Two issues in the modelling of designs that are currently very important in computer-aided engineering research are noted here. The first concerns the amount of time and effort, in particular in embodiment and detail phases, that is spent by engineers first in creating models of the designed artefact and then in creating the further models necessary to evaluate the design and to prepare manufacturing information. Creating these further models from the definition of the design parameters (which may be, but is not always, a geometric representation) often requires much manual effort on the part of the participants. Sometimes the process is less one of alteration or addition to the design model and more one of creating the equivalent viewpoint model from scratch using the viewpoint's native representation method. The need for this high level of effort has had two consequences: firstly, CAD has not delivered the benefits that had been expected, throughout the design process but especially at the concept stage, and, secondly, work on the rapid generation of design models in design automation and in such applications as optimisation has concentrated on high-level design representations such as parametric models. As a result, the search for design representations that serve multiple viewpoints, or that facilitate the automatic or semi-automatic translation between representations has attracted a good deal of research interest in recent years [3][4].

The second issue concerns in particular the concept phase of the design process. In this phase, the effort required to organise the information needed to make judgements about how concept solutions will behave and perform means that concept design is very often concentrated on variants or adaptations of existing designs. The models used in these early design phases reflect this adaptation by being parametric – for example collections of attributes describing the principal dimensions and characteristics of the artefact. For example in commercial aircraft design preliminary estimates of whole aircraft performance and cost can be made from in the order of fifty key aircraft parameters. These estimates predict the performance of the 'new' design based on the known performance of existing designs and the accumulated information related to established product configurations. However, such tools and methods are by their nature only useful for designing aircraft that are very similar to those already in existence. What seems to be lacking is a tool which can evaluate a conceptual design proposal 'without prejudice', i.e. by analysing the design not predicting its performance based on its similarity to existing designs. This is important because the later stages of design process are built upon the results of the conceptual stage.

3. Approaches to Design Representation

A number of approaches to multiple-viewpoint design representations have been investigated, and central among these has been the use of feature-based representations. In feature-based design, models of the designed artefact are represented as collections of features - model elements that have engineering significance [5]. Approaches include:

- Design by features, in which the object model is constructed using features.

- Automatic feature recognition, in which the pre-existing object model is passed through software that identifies the features that are present in the model.
- Interactive feature creation, in which the pre-existing object model is passed through software that allows a human user to identify the features of interest present in the model.

Features allow the designer to associate with the feature model non-geometric information which is available at the design stage and can later be applied to various engineering domains, and features can be used to access information associated with particular feature types during the design process. However, different specialists may choose to use different feature sets for the same engineering part because they assign different engineering meaning to the elements of the part. For example a manufacturing engineer may be interested in the material removed from a part in machining; a structural engineering will be interested in the load bearing material that remains. A single representation that satisfies the needs of each has not been identified, although recent feature research has explored feature definitions, and methods of mapping or converting features, that seeks to overcome this problem of viewpoint dependency of features [3][4].

Features have also been used at the concept design phase to provide conceptual design tools that are less constrained than purely parametric representations. An example of the use of features is seen in ACSYNT (AirCraft SYNThesis) [6] a program produced and developed by NASA in partnership with industry between the 1970s and late 1990s. In this program, aircraft concept models were constructed using features (such as a variety of aerofoil and fuselage elements, wing pylons and so on) containing parameters that linked directly to viewpoint-analyses. This produced a very useful tool; but the architecture of the tool meant that the addition of new analysis routines required significant rewriting of the code to support them. Perhaps more significantly, however, aircraft designs that do not conform to the configurations considered in the original selection of the feature set and analyses are not possible, limiting the value of the tool in concept design.

3.1. Integration of design software systems

The creation and manipulation of design models is largely carried out by the engineers involved in the process. An alternative approach that seeks to automate design evaluation is through the use of integration software that can take the separate computing applications used in a design analysis or evaluation process and mechanize the execution of a sequence of these applications. This is generally done by ‘wrapping’ each separate application with sections of code which deal with its inputs and outputs, and then placing the wrapped segments in a user-defined sequence, the wrapper code dealing with the passing of data between the applications. As applications communicate only through their respective wrappers it is possible to integrate those written in different languages using different file formats etc.

This approach is very useful for known and oft-repeated processes, and is thus excellent for optimisation (including multi-disciplinary), parametric trade studies, sensitivity analysis etc. Integration software using parametric representations [7] can deal well with trade studies because the question of how to analyse the object has been fully answered and the need is to repeat that analysis with slight parametric variations many times. However, such integration is dependent on each separate application being able to 'understand' the description of the design and thus such approaches again mainly depend on parametric descriptions of designs. As a consequence they cannot help with the first run-through of a new process and are of limited use in processes where negotiation between viewpoints is required. Again the need is for design representations that each application can use and that embed the semantic meaning of the design.

4. Implications for Future Design Support

The implications for the design process of these constraints in design representation are profound. That there is a concentration on established designs is clear from industrial practice – for example the basic configuration of gas turbine airliners has not changed significantly in almost 50 years as a quick comparison of the configuration of the Boeing 707 (first flight 1954) and Airbus A340 (1991) can confirm. But there is also a disconnect between the conceptual and later phases of the design process in terms of the approach adopted and the tools used. In the concept phase, in order to explore the design space, representations are needed that may readily be executed and manipulated, and therefore early design is often based on empirical relationships (founded on parametric descriptions of the artefact) that can be executed repeatedly in optimisation or design search. More sophisticated tools are not used because of the difficulty in automating the execution of these tools, because of the manual effort required in setting them up, and because of limitations in data availability. Only in the later stages are more sophisticated analytical tools introduced, but these cannot be used to explore the feasible design space very thoroughly, again because of the effort required in their application.

The danger with the above is that what is produced overall is a refined instance of a sub-optimal concept. With each step in the design process the design space under consideration is reduced - at no stage faster than in the concept phase. The grand question is how to make sure the area of the design space selected at the earliest stage contains the global optimum and not just local optima, and that subsequent design development can find or get close to the global optimum.

This work does not claim to answer this grand question, but seeks to be a part of a move to provide an analytical alternative to semi-empirical concept evaluation methods, which should be of particular use in evaluating concepts that are dissimilar to previous designs. To provide this analytical alternative there is a need for models that allow design automation using more sophisticated tools to be applied early in the design process, and that allow conceptual designs to be explored without such reliance on parametric models and limited feature sets. If successfully created, it is believed that such an approach will also enable a more seamless integration of design representations used at different stages of the design process.

4.1. Choice of demonstration example

The issues of product representation, limited availability of time, data, manpower and resources in industrial conceptual design is mirrored (on a very much smaller scale!) by similar limitations in undergraduate group aircraft design projects, which have the added restriction of limited expertise within the group. Such project work has been chosen to provide an exemplar for the methods being developed by this work.

5. Semantic Mark-Up

Given the above background, the research question that has been addressed is "is it possible to identify CAE modelling techniques that overcome the limitation of existing feature based techniques and parametric approaches, especially in conceptual design?" The method that has been chosen for exploration in this work is the use of semantic mark-up.

5.1. The concept of semantic mark-up

Mark-up is about adding information to an entity to identify its constituents and how that entity may be used. For, example in text documents Goldfarb [8] identifies the purposes of mark-up as "Separating the logical elements of the document; and specifying the processing functions to be performed on those elements."

The first structured mark-up language was IBM's Generalized Mark-up Language, developed in the 1960s by Goldfarb, Mosher and Lorie. This led in the 1970s to the Standard Generalized Mark-up Language (SGML), which was published as an international standard in 1985 [8].

The Hyper Text Mark-up Language (HTML) was developed from SGML and became the standard language for document sharing across the World Wide Web (WWW). In HTML, the document mark-up is principally used to indicate how elements of the document should be presented – for example as headings, tables and so on. Today, other developments of SGML are being used as a tool to create a new kind of WWW, the 'Semantic Web', in which documents are marked up not for display purposes but so that they can be intelligently processed automatically. The Semantic Web is an attempt to "…bring structure to the meaningful content of Web pages"[9], to allow computers to manipulate and deal with the contents of the Web on a more sophisticated (abstract) level. The objective of the work reported here is to do the same for CAE models.

To allow computers to manipulate documents as more than symbols, but instead with regard to their meaning (semantic content), documents to be used on the Semantic Web are 'marked-up' using new mark-up languages, including the eXtensible Mark-up Language, XML, and the Resource Description Framework (RDF) [9] to make the semantics or meaning of the documents visible to the computer. However, in order for the computer to 'understand' the expressions of meaning such that it can manipulate the documents 'intelligently' a third element is needed. RDF makes the 'meaning' visible to the computer but it needs rules and structure for manipulating it. In the context of the Semantic Web these are provided by ontologies. An ontology in this context is a structure that defines the place of every entity in a domain and the relations between entities allowing inferences to be made between them.

5.2. Semantic mark-up for CAE

The Semantic Web is an attempt to solve the problem of computers only being able to represent and manipulate documents at a very low level of abstraction. Essentially a text document can be represented as a collection of characters, but the computational ability to manipulate it is limited because a computer only 'understands' the document as a collection of symbols. In a similar way computers can represent a design object but their ability to manipulate it as a collection of largely geometric entities is limited.

XML, RDF and ontologies allow computers to manipulate documents at a higher level of abstraction to some degree. Can equivalent technologies to those of the Semantic Web be developed to allow the computer to deal with CAE models more 'intelligently'?

For the most part, a geometric model in a modern CAD system will be represented as a collection of faces, edges and vertices. If the CAD software stores a construction history or uses features, then further layers of structure may be encoded in addition to the boundary representation (B-rep) model, but often features are just used for the purposes of rapid construction and are not persistent, and construction history contains no information about the engineering significance of the model.

Use of technologies in CAD that are similar to the Semantic Web technologies of XML and the RDF may allow another higher layer of abstraction to be coded into CAD models. By using appropriate mark-up, information relevant to manufacturing and analysis may be embedded into the CAD models. And, just as ontologies allow the marked-up documents to be manipulated more intelligently by computer, so ontologies in CAE would allow the definition of engineering entities and methods to allow 'marked-up' CAD models to be automatically converted into viewpoint-dependent models for analysis and manufacturing. These ontologies need to define the entities in the domain (of the analyses and other applications) the relationships between entities (including pieces of process that manipulate the entities) and rules of 'inference' (process sequences) for moving between the entities.

5.2.1. How will the model be marked-up?

In this context mark-up essentially involves associating elements of geometry with non-geometric information. This is what features allow, and therefore the concept of features is a natural starting point for finding a method to mark-up the model. However, it is important to make clear that the mark-up concept is about more than simply identifying geometric features; fundamentally it is about increasing the amount of information encapsulated in a model to aid downstream analysis.

It is argued that all three established feature creation methods, outlined in section 3 (design-by-features, automatic feature recognition, interactive feature creation) have potential for CAE model mark-up, and that future CAE systems will embed all approaches.

Design-by-features (DBF) might be used in a conventional way, with the entities associated with each feature being marked up as such. DBF would ensure comprehensive mark-up of the created model but would dictate the method of model creation, including the software that would have to be used, and limits the range of geometry that can be created. Any new evaluations would necessitate changing the feature set. Nevertheless, design-by-features would ensure rapid model generation and correct mark-up of the models.

Automatic feature recognition (AFR) may be used to process the geometric model and to label those entities corresponding to features and other meaningful parts of the model. AFR may also limit the mark-up to features that could be recognised, and may have trouble differentiating between geometrically or topologically similar but functionally different features e.g. engine intake and exhaust. However, automatic recognition allows easier expansion of the evaluations covered as the model may be passed through the recognition software any number of times to look for different feature sets.

Interactive feature creation (IFC) allows labels to be attached to a model by the user interactively selecting entities for mark-up. IFC ensures correct mark-up of the model according to the user's intent, and only requires the features of interest to be addressed. However, for models with many features of interest this would be a laborious approach, and there is a greater risk of error than with the other techniques.

These differences may be exploited for different applications. For variant design, a design-by-features type approach including the mark-up in the building of the model may offer the best solution, whereas with novel concepts and early design where features of interest are generally fewer in number interactive creation may serve better.

Most design work is neither wholly variant nor wholly novel, supporting the view that a combination of the approaches is most appropriate. For instance in the creation of the model, variant features may be created and marked-up using the design-by approach. The model would then be interactively marked-up for novel features and information requiring a higher understanding of the object and/or design intent. An automatic recognition approach may be used in between to reduce the number of features of interest requiring interactive mark-up, or at the end to add further information that can be derived or inferred from the information present e.g. if the material density of a part is added interactively then from this and the existing geometry the mass, centre of mass, moments of inertia etc. of the part could be automatically calculated and appended to the model (as mark-up), ready for use in further analyses.

For all of the approaches the first technical step in creating the method, and subject of our current research, involves finding the best way to mark-up information in the geometrical representation of the model. The approach that we are taking is for user defined objects (UDO's i.e. user-defined data structures) to be attached to elements of the B-rep. These can be attached at whatever geometric level is most appropriate – for example identifying the external boundary of a model would involve labelling all faces, the part density can be attached at the solid level, faces can be labelled as being subject to loads and manufacturing features can be generally indicated by the labelling of faces and edges (e.g. a cylindrical surface and a centre-line for a hole).

5.3. Example application: undergraduate group aircraft design projects

Two exemplar demonstrations are currently being developed. The first involves marking up simple parts from two example perspectives (e.g. stress analysis and manufacturing) and then using that mark-up to automate the processing of the model. The second is in support of design evaluation of conceptual aircraft models in undergraduate group design projects. We concentrate here on the latter.

Figure 1(a) is a representation of a typical undergraduate group design process from concept generation to final design proposal. It does not seem suitable to aim the method at the concept selection stage in this educational setting as this stage tends to be abbreviated in student design projects with the selection of baseline configuration made on quantitative and subjective judgements [10]. There may be several reasons for this, including the relatively short duration of such projects (of the order of 3 months), and it must be remembered that the prime concern of the project is not to produce the best possible solution aircraft for the brief, but to provide the students with experience of group design. Limiting the designs chosen to known patterns allows the educators to concentrate on the process rather than having to find suitable analysis methods for unusual configuration concepts. The approach described here thus comes into play when a baseline configuration has been selected for evaluation.

With the baseline configuration selected and first estimates of prime dimensions and parameters in place, it is normal for an initial 3-view drawing of the aircraft configuration being studied to be made. At this stage that sketch is translated into an initial 3D CAD model, which would then be marked-up for translation to models for various evaluations, leading to fully or semi-automated production of the evaluations.

Current hand calculation methods that are normally used for the analyses at this stage, and indeed throughout the length of the project, require much time and effort from the student just to perform the calculations to produce the refined parameter values. This leaves little time for iteration, or exploring the sensitivities (design space) of the design, or for experience of the negotiation that is so important a part of design.

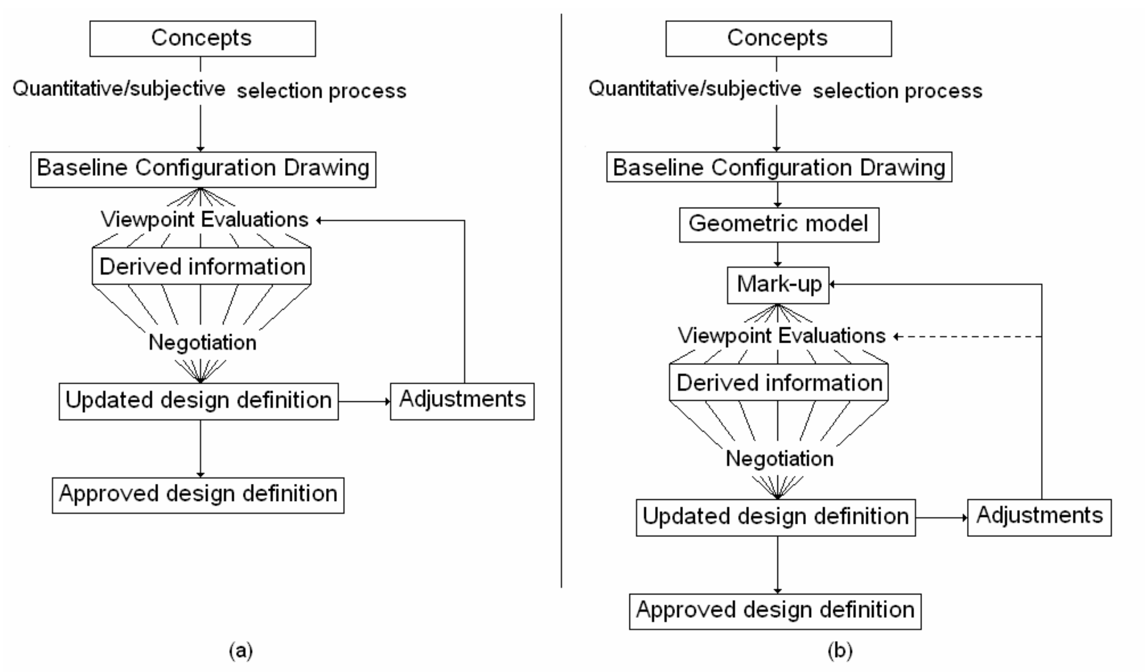

Figure 1. Undergraduate aircraft design process

Figure 1(b) is a representation of what the process looks like with a mark-up method included. The dotted arrow shows the ideal arrangement where the mark-up does not need to be re-done as the aircraft definition changes, however, just how much it will be possible to change the central product model before the original mark-up is rendered out-of-date will depend on how mark-up is achieved.

5.4. Marking up the model

Since the ultimate purpose of the example application is educational this has some influence on the choice of approach to mark-up. If it was possible to create an automatic recognition routine that could reliably mark-up an aircraft model (across a wide configuration range) this would not teach the student much if all it required them to do was run their model through the routine. Design-by-features might have the same problem and further restrict the configuration ranges that can be considered.

If interactive feature creation is used the student must learn what features of the geometry are important for different design evaluations and selectively mark them up appropriately. With interactive feature creation the properties of a feature can be less strictly tied to geometry, allowing features with very different geometry but requiring the same treatment with regard to an analysis to be identified using the same feature definition. This combined with allowing the geometry to be built by whatever method the user prefers makes the interactive feature creation technique seem the most promising at this stage

5.5. Description of operation

This section briefly outlines the current approach to operation of the method:

The student begins by translating the initial layout sketch into a CAD model. The CAD model will then be opened in a mark-up program developed using the CAD system's API. This program leads the student(s) through the process of marking up analysis-significant features of the model, adding any necessary extra information at the same time.

The output representation comprises geometric features marked up with further information such as indications of external boundaries, principal axes, surfaces with FEA boundary conditions and so on. The exact form of the output representation from this program will depend on current investigations of the capabilities of the chosen CAD software's API and output file formats. Currently, we are exploring three possible routes:

1. Non-geometric information and associations with and between geometry held within the CAD system's database as attributes on the system's B-rep entities.
2. Geometry held in the CAD system's file format with a separate file containing the rest of the information including cross-referencing identifiers to the relevant geometric entities in the CAD file.
3. Geometry associations and mark-up information coded using a cross-software standard representation such as STEP or the emerging CAD Services standard.

Option 1 requires using CAD system file formats in a way that they were not originally intended to be used. However, conceptually all that is required is that the association of user-defined attributes with CAD system entities is allowed. The interpretation of these is then left to the receiving program.

Clearly option 2 is more unwieldy, but the most likely to be feasible as it does not demand much from the existing file formats, only that they make available persistent identifiers for geometric entities within a part file. However, preliminary investigation suggests that persistence of identifiers in some systems is more of an issue than might be expected.

Option 3 would have many advantages in making the method immediately applicable to other CAE systems by utilising a format they are already equipped to accept. However, it would not remove the need for further work within their API's in order to allow them to use the information contained in the marking-up.

Also important from an implementation point of view is the way that ontological data are defined and stored. This requires a mechanism whereby information about the way entities are marked-up and manipulated may be defined and stored. At present this information is being embedded in the experimental programs that we are writing to test the ideas, but in practical applications a more transparent way of defining and applying this information is required. The aim is to produce a working example that demonstrates the principle of 'semantic' mark-up of CAD models for automatic viewpoint translation, not an industry-ready tool.

The marked-up model should contain all the information necessary to perform a design evaluation or range of evaluations. This should allow a much higher level of automation of the process of building the evaluation models, and thus a reduction in the time taken to build the model and therefore more time for iterations of the design process and to spend on interpreting and exploring the evaluation – more time to explore the design space from that particular viewpoint.

In the context of the exemplar that means the students should be able to develop a much better understanding of their appointed discipline's view of the design (its sensitivities and ideals), enabling more informed negotiations between the viewpoints.

6. Conclusions

Within the current spectrum of computer-aided design methods and tools there is a concentration on aiding the detail design stages and variant design. This has lead to the conceptual design stages and novel design being far less well supported than might be hoped, with the result that conservatism is encouraged in engineering design.

This work aims to develop and demonstrate a method for reducing the manual workload required to produce engineering viewpoint analysis models from a central geometric design model for multiple viewpoint design. The method being developed is conceptually similar to the concept of semantic mark-up of documents for the semantic web. The idea is that by marking-up a geometric design model with analysis process and other design evaluation information and creating a structure analogous to an ontology in the Semantic Web the production of the analysis models may be automated. In this paper we describe how this approach would allow design by features, feature recognition and interactive feature creation to be combined into a unified approach.

There are many issues remaining to be solved, in particular how ontological data may be specified to describe how the design entities may be manipulated in different evaluations, and how non geometric information, for example concerning product structure, may be integrated. Nevertheless, it is hoped that by applying ideas of semantic mark-up in design, a foundation may be laid for a step forward in design representation and automation.

References

[1] Lee K H and McMahon, CA, "Characteristics of a multi-viewpoint feature-based design automation environment", Proc ICED03, Stockholm, August 2003.

[2] Suh, N. P. "The Principles of Design", Oxford University Press, New York, 1990.

[3] Bronsvoort, W.F., Noort, A., van den Berg, J., and Hoek, G.F.M., "Product development with multiple-view feature modelling", CD-ROM Proceedings of the IFIP Conference on Feature Modelling and Advanced Design-For-The-Life-Cycle Systems, 12-14 June, Valenciennes, France, 2001.

[4] Hoffmann, C.M., and Joan-Ariyo, R., "Distributed Maintenance of Multiple Product Views", Computer-Aided Design, Vol. 32, 2000, pp.421-431.

[5] Shah J J and Mäntylä, M, "Parametric and feature-based CAD/CAM: Concepts, Techniques, Applications ", John Wiley and Sons, Inc. New York, 1995.

[6] Myklebust A and Gelhausen, P, "Putting the ACSYNT on Aircraft Design", Aerospace America, 32 (3), 1994, pp.616-622.

[7] Dabke, P., Cox, A., and Johnson, D., "NetBuilder: an Environment for Integrating Tools and People", Computer-Aided Design, 30, 1998, pp.465-472.

[8] Goldfarb, C F, "The SGML Handbook", Clarendon Press, Oxford, 1990.

[9] Berners-Lee, T., Hendler, J. & Lassila, O., "The Semantic Web", Scientific American, 284 (5), 2001, pp.34-43.

[10] Jenkinson, L.R., & Marchman III, J.F., "Aircraft Design Projects for Engineering Students", Butterworth-Heinemann, 2003.

Daniel Davies
University of Bath
Department of Mechanical Engineering
Bath
BA2 7AY
England
Phone: (01225) 384166
Email: en8dd@bath.ac.uk